iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** VII **Month of publication:** July 2023

DOI: https://doi.org/10.22214/ijraset.2023.55092

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VII Jul 2023- Available at www.ijraset.com

LSTM-based Stock Price Prediction by Utilizing an Adaptive Labelling Strategy to Optimize the Feature Selection Duration

Rayaan Kakad[1], Sudhir Shinde[2]
[1]*Jamnabai Narsee International School*
[2]*Indian Institute of Technology Bombay*

Abstract: ***The growing popularity of predicting stock prices using advanced machine learning techniques, particularly Long Short-Term Memory (LSTM) models, has been driven by their ability to uncover complex patterns that are challenging for humans to identify. This trend has been supported by the widespread availability of data. In this study, we employed an adaptive labelling strategy to maximize intraday returns and examined the impact of the quantity of training data points on the accuracy of LSTM-based predictions of intraday returns. We evaluated the average returns and volatility across different training spans, verifying the results over an extended period. By comparing various partition sizes, we determined that a partition size of 240 yielded high Sharpe ratios (>2) and improved mean intraday returns. Our research demonstrates that an adaptive labeling strategy combined with LSTM models can be instrumental in achieving maximum intraday returns, with a partition size of 240 being the optimal choice for predicting stock prices.***
Keywords: LSTM, intraday returns, partition size, training span, labelling.

I. INTRODUCTION

In recent years, a surge of interest in stock trading, with many countries experiencing a rise in trading accounts during the Covid-19 pandemic. With the increasing volatility of the stock market, there has also been a growing interest in stock price prediction. The availability of data and the development of deep learning models have fueled this interest. Statistical models like the autoregressive integrated moving average (ARIMA) were initially used to forecast stock prices [1]. However, these models were ineffective due to the non-linear nature of stock market growth and decline. In recent years, researchers have turned to computational intelligence and artificial neural networks (ANNs) to develop non-linear models for stock market forecasting [2,3].

The Long Short-Term Memory (LSTM) model is a popular type of Artificial Neural Network (ANN) used for time-series forecasting. It falls under the category of Recurrent Neural Networks (RNNs) and is tailored to handle sequential data. It comprises four units namely cell, input, forget, and output - and can select which data to consider and which to ignore for each instance [4]. LSTM models are particularly effective at identifying patterns that occur after irregular intervals, which can be difficult for human intelligence to detect. This is due to their ability to avoid the vanishing gradient problem during backpropagation, giving them an edge over other models like hidden Markov models and classic RNNs. Overall, the development of LSTM models and other deep learning techniques has revolutionized the field of stock market forecasting and has the potential to help traders make more informed investment decisions [5,6]. The key advantage of LSTM is its ability to store and access information over long periods, making it particularly useful for tasks involving sequential data with long-term dependencies. The duration for feature selection is a challenging task in stock prediction with high accuracy. Historically, researchers have employed one-year partitions, which corresponds to roughly 260 days, to identify variables with greater weightage [7]. This method is effective for seasonal forecasting or industries with extended business cycles. However, it may not be the most suitable approach for industries characterized by fluctuating public demand and dynamic business scenarios. Therefore, determining an optimized period size for dividing the training and feature selection spans becomes crucial. The literature review revealed various researchers have used different strategy to compute the average intraday return [5-8]. However, it appears that an adaptive labeling strategy has not been utilized to calculate intraday return. Hence, this study utilizes an adaptive labeling strategy when training the LSTM model to obtain maximum intraday return of the 47 stocks of the Nifty 50 group. An optimize period size is also determined to divide training and feature selection span with higher return and lower volatility. Further, mean return, standard deviation, and Sharpe ratio are evaluated. Additionally, a comparison is made with the 50% selling and 50% holding strategy.

A. LSTM Algorithm

The LSTM architecture consists of input shape i.e. no of days data need to be taken into consideration. The 25 LSTM units is used in the model. Each LSTM unit has three gates namely input, forget, and output that regulate the flow of information and determine what information to keep or discard at each time step. Further, a fully connected output layer with softmax activation function is used. The categorical cross-entropy loss function is used during training. We've also used RMSProp optimizer, which is a popular choice for training LSTM.

To calculate mean return using an LSTM algorithm, following formula is used [9]:

$$\text{Mean Return} = \frac{\text{Ending Price} - \text{Starting Price}}{\text{Starting Price}}$$

where Ending Price is the predicted price at the end of the testing period and Starting Price is the actual price at the beginning of the testing period. Metrics like mean squared error (MSE) and root mean squared error (RMSE) are utilized to assess the performance of the model. This will give an idea of how accurate the model is in predicting stock prices.

To calculate the standard deviation from mean return, the following formula is used [10]:

$$\text{Standard Deviation} = \sqrt{\sum \frac{(\text{predicted prices} - \text{actual prices} - \text{mean return})^2}{n-1}} S$$

where n is the number of data points.

Predicted results may consist lot of fluctuation in the results. Therefore, it necessary to calculate risk adjusted return. The Sharpe ratio is a measure of risk-adjusted return and is calculated as follows [11]:

$$\textit{Sharpe ratio} = \frac{\text{Excess Return}}{\text{Standard Deviation}}$$

While the Sharpe ratio is a valuable metric for assessing investment performance, it has certain limitations, and a comprehensive analysis should incorporate other metrics as well. It's worth noting that a higher Sharpe ratio indicates better investment performance relative to risk.

II. METHODOLOGY

To prepare the data for the present study, Nifty50 stocks spanning the period from 2013 to 2019 are chosen. The data was pre-processed by dropping any days for which data was not present and extracting the open and close values for each stock. Feature scaling was performed using a robust scaler. When dealing with a large number of input variables, feature selection becomes crucial in building predictive models. This is because a high number of input variables can lead to increased computation costs and introduce randomness into the model, as noted in [8]. As a result, we have employed feature selection techniques to identify variables with higher influence and reduce the number of inputs. This helps to improve the model's accuracy while reducing computation time and complexity. The process of stock prediction using LSTM typically involves three primary steps: feature generation, training, and testing. During feature generation, a set of relevant features and important variables are selected. In order to find the optimal partition size for the training phase, the training span is divided into partitions of 50 days. This corresponds to approximately 5-6 partitions per year, based on an assumption of 260 trading days per year. The prediction accuracy is then assessed for various partition sizes, usually five, resulting in varying numbers of partitions per year.

The data labelling process involves marking the data points based on their return values. The labelling strategy used here is such that if the return value is less than zero, it is labelled as zero, whereas if the return value is greater than zero, it is labelled as one. More specifically, the intraday return is used for labelling, with positive intraday return marked as 1, and negative intraday return marked as 0. This labelling process is explained in the flow chart (refer Fig. 1). To achieve this, an adaptive labelling strategy is used to identify stocks that are likely to yield a profit in intraday transactions. To evaluate the performance of different partition sizes in the duration, the number of training data points is varied, and their impact is measured using Sharpe ratios, returns, and reduced volatility. The partition sizes that yield better returns and reduced volatility are deemed appropriate for the analysis.

To build an LSTM model for stock prediction, a Python library Keras is used. The model should consist of an input layer, one or more LSTM layers, and an output layer. After building the model, it needs to be trained using the training data. During the training process, the model learns how to predict future stock prices based on past data. Once the model is trained, it can be tested using the testing data, and the mean return can be calculated based on its predictions.

The accuracy of the model can be measured by calculating the daily returns, standard deviation, and Sharpe ratio on the test data. The approach involves using an adaptive labelling strategy for stocks that would return a profit in the intraday transaction. The figure 1 shows a flow chart explaining the end-to-end model training from data pre-processing, data labelling, model training, and accuracy calculation.

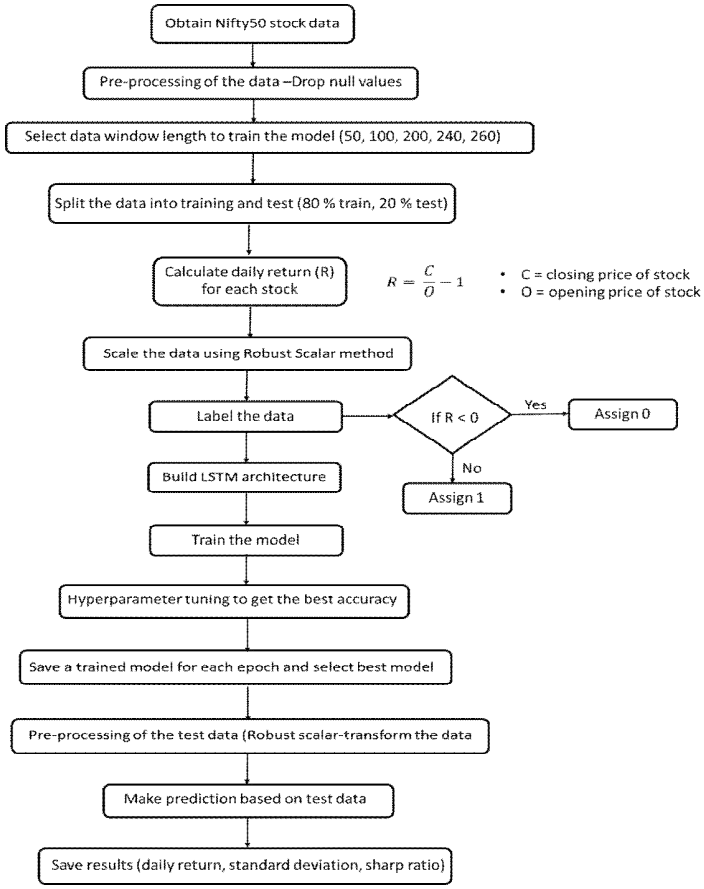


Figure 1 Algorithm to obtain daily mean return, standard deviation, and Sharpe ratio

III. RESULTS AND DISCUSSION

The provided data represents the average returns for different partition day sizes from 2013 to 2019. Throughout the years and partition sizes, the returns fluctuated between positive and negative values. In 2018, the highest mean return was observed with a partition day size of 260, while the lowest mean return occurred in 2016 with a partition day size of 100. Interestingly, the average return in the initial year, 2013, was positive, but it decreased in 2014. However, in 2018, the mean return surpassed that of 2019, except when the partition day size was 240. This suggests that increasing the number of days in the partition positively impacted the returns, as the LSTM model can better capture longer sequences of data. It is worth noting that as the data window length (number of days) increased, the model became more complex and computationally intensive to train.

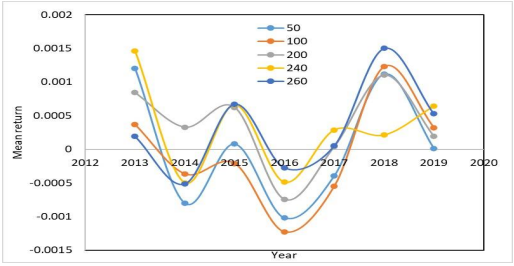


Figure 2 Variation in mean return with year at different value of partition day size

Figure 3 displays the standard deviation for five different partition day sizes across the years 2013 to 2019. The standard deviation remains relatively stable within a range of 0.4%, except for certain instances, such as the 100-partition day size in 2013 and 2019, the 240-partition day size in 2017, and the 200-partition day size in 2014. These exceptions indicate a higher level of variability in those cases. The variation might be caused because of the events happened in for a short duration and LSTM model with short no. of days for training might not be able to capture the variation. The LSTM models are designed to capture patterns over time. When the data exhibits a high degree of variability, such as when brief events occur, it can be challenging for the model to accurately capture these patterns. The length of time that the LSTM model has been trained for can also impact its ability to capture the variability in the data. When the model is trained on a shorter period, it has less exposure to the data and may not have had sufficient opportunity to learn the patterns that are present in the data. Therefore, if the variability in the data is caused by brief events, it may be more difficult for a model trained over a short period to accurately capture this variability. In summary, the variability in the data may be challenging to capture using an LSTM model trained over a short period of time, particularly if the variability is caused by brief events.

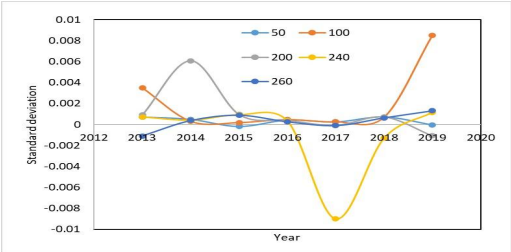


Figure 3 Variation in standard deviation with year at different value of partition day size

Figure 4 depicts the variation of the Sharpe ratio across different years and partition day sizes. The trend in the Sharpe ratio closely follows that of the mean return, indicating a linear dependency between the two. The highest Sharpe ratio was observed in 2018, specifically with a partition day size of 260, while the lowest Sharpe ratio occurred in 2016 with a partition day size of 100. This suggests that higher mean returns generally correspond to higher Sharpe ratios, indicating a better risk-adjusted performance.

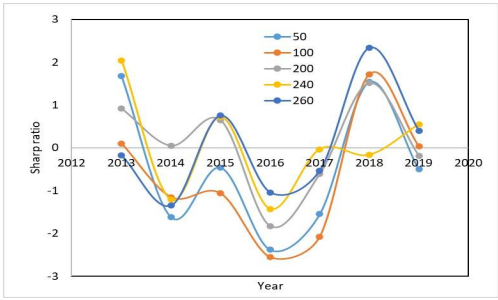


Figure 4 Variation in Sharpe ratio with year at different value of partition day size

Figure 5 shows comparison of adaptive labelling strategy and 50-50 strategy for mean intraday return at five different value of partition day size. For a higher partition day size (200, 240, 260), huge improvement in mean intraday return is observed in the case of adaptive labelling strategy compared to 50-50 strategy. This is mainly due to increase in amount of training data which captures the fluctuation of the data more accurately.

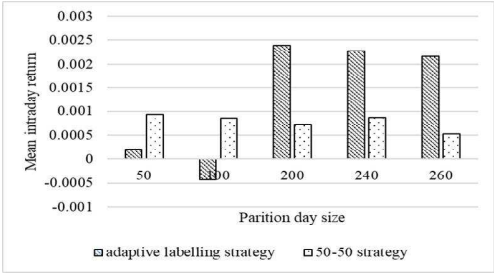


Figure 5 Comparison of adaptive labelling strategy and 50-50 strategy for mean intraday return

IV. CONCLUSIONS

In the current study, an adaptive labelling strategy was employed in conjunction with the LSTM algorithm to maximize intraday returns. The Nifty 50 data from 2013 to 2019 was pre-processed and labelled using this strategy. Subsequently, the model was trained on different numbers of partition days. Mean intraday returns, standard deviation, and the Sharpe ratio were calculated based on the trained model.

The results revealed that increasing the number of partition days led to higher returns, as the LSTM model demonstrated its ability to capture longer sequences of data. However, it is important to note that this also introduced greater complexity to the model and made it computationally intensive to train. Among the various partition day sizes examined, a partition day size of 260 exhibited the minimum standard deviation. This can be attributed to the advantages gained by the model in terms of capturing long-term patterns and reducing the impact of short-term fluctuations. Furthermore, the adaptive labelling strategy employed in this study showcased significant improvement in mean return compared to the 50-50 labelling strategy. This indicates the effectiveness of the adaptive approach in generating more favourable outcomes. Overall, the study highlights the benefits of utilizing an adaptive labelling strategy in conjunction with the LSTM algorithm, particularly when considering longer partition day sizes, to enhance intraday returns and manage risks more effectively.

A. Declaration Of Competing Interest

The authors have no competing interests to declare that are relevant to the content of this article.

B. Ethical Approval

This work does not contain any studies with human or animal subjects performed by any of the authors.

C. Funding Source

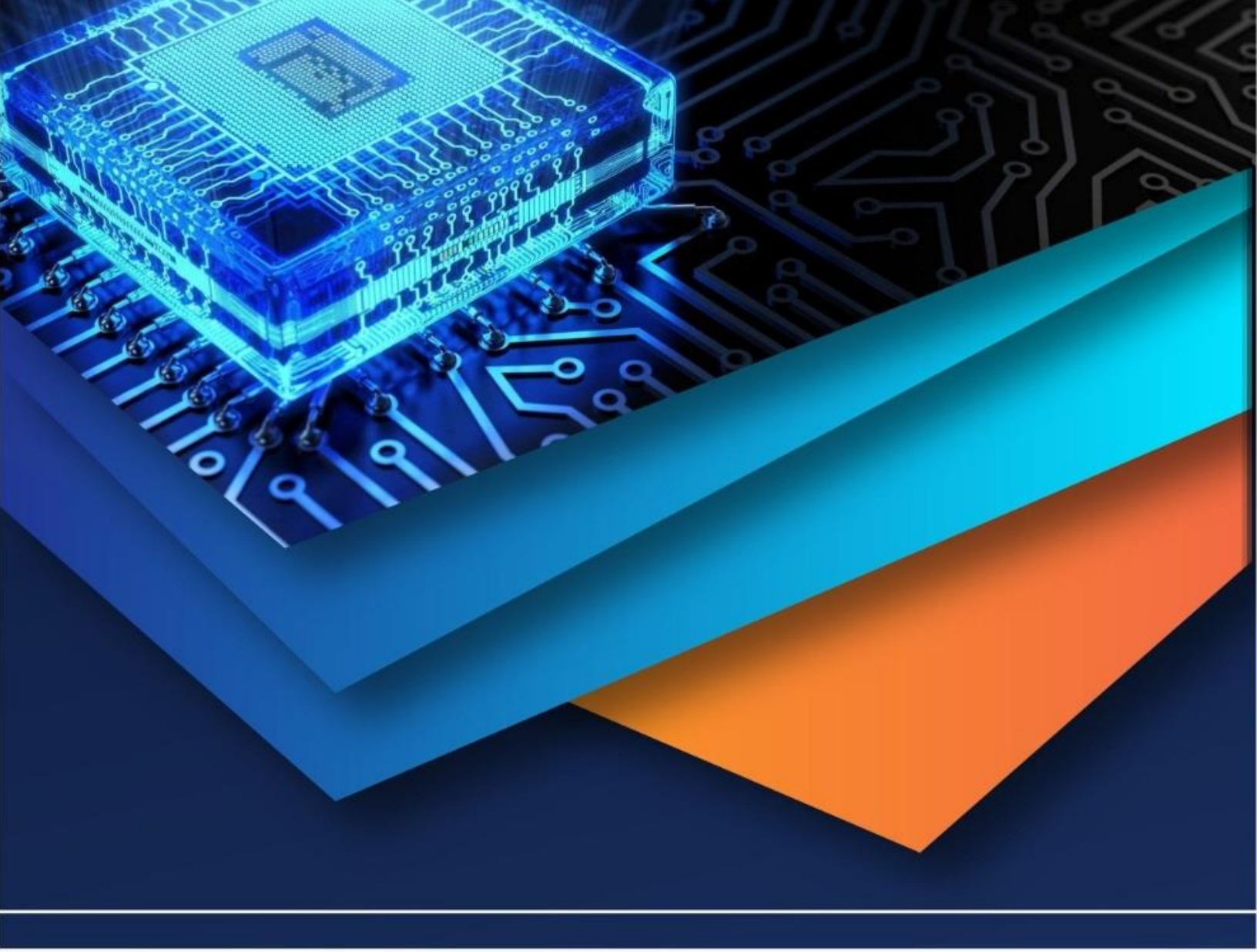
No funding was received for this study.

D. Data Availability

The data that support the findings of this study are available on request from the corresponding author.

REFERENCES

[1] Fischer, T. and Krauss, C., 2018. Deep learning with long short-term memory networks for financial market predictions. European Journal of Operational Research, 270(2), pp.654-669.
[2] Siami-Namini, S. and Namin, A.S., 2018. Forecasting economics and financial time series: ARIMA vs. LSTM. arXiv preprint arXiv:1803.06386.
[3] Ghosh, P., Neufeld, A. and Sahoo, J.K., 2021. Forecasting directional movements of stock prices for intraday trading using LSTM and random forests. Finance Research Letters, p.102280
[4] Sharma, A., Tiwari, P., Gupta, A. and Garg, P., 2021. Use of LSTM and ARIMAX Algorithms to Analyze Impact of Sentiment Analysis in Stock Market Prediction. In Intelligent Data Communication Technologies and Internet of Things: Proceedings of ICICI 2020 (pp. 377-394). Springer Singapore.
[5] Cheng M, Cai K, Li M. Rwf-2000: An open large scale video database for violence detection. In2020 25th International Conference on Pattern Recognition (ICPR) 2021 Jan 10 (pp. 4183-4190). IEEE.
[6] Rybalkin, V., Sudarshan, C., Weis, C., Lappas, J., Wehn, N. and Cheng, L., 2020. Efficient Hardware Architectures for 1D-and MD-LSTM Networks. Journal of Signal Processing Systems, 92(11), pp.1219-1245.
[7] Das, S. and Mishra, S., 2019. Advanced deep learning framework for stock value prediction. International Journal of Innovative Technology and Exploring Engineering, 8(10), pp.2358-2367.
[8] Sumon, S.A., Shahria, T., Goni, R., Hasan, N., Almarufuzzaman, A.M. and Rahman, R.M., 2019, April. Violent Crowd Flow Detection Using Deep Learning. In ACIIDS (1) (pp. 613-625).
[9] Rybalkin, V., Sudarshan, C., Weis, C., Lappas, J., Wehn, N. and Cheng, L., 2020. Efficient Hardware Architectures for 1D-and MD-LSTM Networks. Journal of Signal Processing Systems, 92(11), pp.1219-1245.
[10] Siami-Namini, S. and Namin, A.S., 2018. Forecasting economics and financial time series: ARIMA vs. LSTM. arXiv preprint arXiv:1803.06386.
[11] Qiu, J., Wang, B. and Zhou, C., 2020. Forecasting stock prices with long-short term memory neural network based on attention mechanism. PloS one, 15(1), p.e0227222.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY